Absolute Stability of Chaotic Asynchronous Multi-Interactions Schemes for Solving ODE

Pascal Redou, Laurent Gaubert, Gireg Desmeulles, Vincent Rodin, Pierre-Antoine Béal, Christophe Le Gal

Absolute stability of chaotic asynchronous multi-interactions schemes for solving ODE

Pascal Redou*, Laurent Gaubert*, Gireg Desmeulles*, Vincent Rodin†, Pierre-Antoine Béal‡ and Christophe Le Gal‡
*UEB, Enib, LISYC- EA 3883, CERV
Email: redou@enib.fr
†UEB, UBO, LISYC- EA 3883
‡UEB, LISYC- EA 3883, CervVal

***Abstract*—Multi Interaction Systems are dedicated to real-time interactive simulations. They are based on chaotic and asynchronous scheduling of autonomous processes, in which physical or biological phenomena involved in the system are desynchronized. This allows interactivity, especially the capability to add or remove phenomena in the course of a simulation. This "desynchronized scheduling" leads to methods of resolution of ordinary differential systems and partial derivative equations. Proofs of convergence for these methods have been given, but the problem of absolute stability, eventhough it is crucial when considering multiscale or stiff problems, has not yet been treated. The aim of this article is to present absolute stability conditions for chaotic and asynchronous schemes. We give criteria so as to predict instability thresholds, and study in details the significative example of a damped spring-mass system. Our results, which make use of random matrices products theory, stress the point that the desynchronization of phenomena, and a random scheduling of their activations, can lead to instability.**

***Index Terms*—Chaotic asynchronous scheduling, Multi-interaction systems, Ordinary differential systems, Absolute stability, Random matrices products.**

I. Introduction

Multi Interaction Systems (MIS) [1] were introduced in the context of Virtual Reality. The initial aim was to provide medical researchers with a simulator dedicated to virtual experimentation and allowing the user four essential points:

1) Interact with the simulated system in the course of the simulation, without stopping it, by adding or removing interactions or constituents in the system, so as to be as close as possible from the *in vitro* experimentation.
2) Achieve this interactive simulation without knowing anything about programmation or numerical methods for solving differential systems.
3) Take into account widely different time and space scales for simulated phenomena.
4) Obtain as precise results as possible, particularly when solving differential or partial derivative systems.

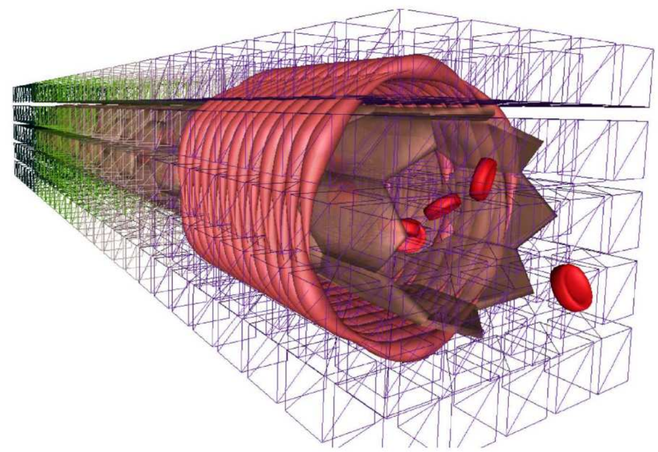

Fig. 1. In Virtuo simulation of endothelium using a MIS.

Thereby, the challenge was to create a simulator for what was called *in virtuo* experimentation [2], that is, to summarize, *in silico* computations in the conditions of *in vitro* experiments. For this purpose, the MIS paradigm proposed to reify interactions into the system instead of constituents, with the main and basical advantage to provide modularity, i.e. adding or removing interactions in course of simulation. Thus, a MIS can be seen as a collection of autonomous processes-interactions, each acting on a collection of variables-constituents, and carrying its own time step. This radical change in perspective has made feasible the constraints, outlined above, of *in virtuo* experimentation. It also led to the choice of a new kind of simulation algorithms, based upon random scheduling of interactions inside the system: chaotic asynchronous scheduling [1], [3], [4]. The principle is to consider each (physical, biological) phenomenon acting on the system -i.e. an interaction between constituents- as autonomous. The simulation engine evolves interactions asynchronously (one after the other, into cycles) and chaotically (the order of interventions changes randomly from one cycle to the other). This scheduling was chosen in order to avoid the typical inflexibility of synchronous systems, as well as bias in numerical results.

From a formal point of view, chaotic asynchronous scheduling provides methods of resolution of ordinary differential equations or systems (say, to simplify, ODE) [3], as well as methods for partial derivative equations [4]. The present work deals with the case of numerical resolution of ODE. Let us give the principle of chaotic asynchronous scheduling in this context: if one wants to solve the cauchy problem

$$Y'(t) = (f_1 + \cdots + f_p)(t, Y(t)), \quad Y(t_0) = Y_0 \qquad (1)$$

the principle is to consider functions f_i as autonomous agents, what is necessary when desynchronizing the different phenomena represented by each of these functions. Considering a numerical method for solving (1), the matching chaotic asynchronous method will be given by successive applications of the chosen method, one for each function. These resolutions take place during the same time step, and the order of resolutions, that is, the order of interventions of functions/phenomena f_i, changes randomly at each time step. Details about this process are given in section III.

This desynchronization eases a modular and incremental building of the numerical model. This is especially usefull when building biochemical models, since the modeller usually selects, subjectively, the reactions which are most likeky involved, and runs the model. If results are not correct enough, the model is incremented with other reactions, etc., until a satisfying model is obtained. Modularity makes this process natural and doesn't require to stop the simulation to modify the code of equations.

Furthermore, chaotic asynchronous simulation provides a means to bear with non-determinism, which occurs most of the time in chaotic systems because of causality between phenomena at the beginning of the experiment, at a very small scale [5]. Introducing random causality inside a computation time step facilitates the construction of simulators able to report a non-determinist behavior.

Many applications have been achieved in different domains, though, as said above, biochemical kinetics is a natural application context: a classical example is given by cancer, since chromosomic instability [6] implies on a regular basis modifications or creations of new reactions [7]. In this context, an application of this scheduling to computer simulation of multiple myeloma was recently achieved [8]. Notice that it is also used for simulation of MAPK pathway [9], and simulation of the extrinsic pathway of blood coagulation [10]. In an other context, chaotic asynchronous scheduling is used for simulation of sea states, which is typically multi-model and multi-scale [11].

The convergence of these methods has been established [3], [4], but the problem of absolute stability [12] has not yet been treated, despite its importance: indeed, the region of absolute stability can be seen as the set of values of the time step outside which the distance between the exact solution and the approximate gets out of control. Thus, when simulating multiscale problems, one has to find a compromise between precision and a realistic time simulation, and this choice can not be made without knowing the region of absolute stability. Another important case where this knowledge is crucial is given by stiff problems [13], with brutal variations of the solution of an ODE. The aim of this article is to present absolute stability conditions for chaotic and asynchronous schemes. We give general results, based upon the theory of products of random matrices, and stress the point that in certain circumstances, these schemes may impose strong conditions on the time step, mainly when opposing forces are at work in the system. A significative illustration is the case of a damped spring-mass system where the different physical phenomena are desynchronized.

In section II, we remind the reader of the problem of absolute stability of methods for solving ODEs, so as properties of classical explicit and implicit schemes. In section III, we describe how desynchronization of phenomena leads to define asynchronous and chaotic asynchronous schemes. We also recall results of convergence for these methods. Sections IV and V expose the main results of this paper : we study absolute stability for asynchronous and chaotic asychronous schemes, in a general context, providing conditions on integration steps. Finally, section VI exposes the practical example of a damped spring-mass system, where the three phenomena involved are not considered as synchronous. In this case, we show that absolute stability conditions can be drastical.

II. Absolute stability issues

Let us first remind the reader of absolute stability issues [14], [13], so as of classical cases.

A. Definitions

We consider the following differential system (classically named *test equation*)

$$X'(t) = A \cdot X(t) \qquad (2)$$

where A is a square matrix with *distinct eigenvalues all lying in the negative half-plane* $\Re(z) < 0$. Its general solution is

$$X(t) = \exp(tA) \cdot X(0)$$

One has, under these conditions,

$$\lim_{t \to \infty} X(t) = \vec{0}$$

Consider the one dimensional case $y' = \lambda y$, $\Re(\lambda) < 0$, and assume that, with the method which is used, y_n approximates the exact solution $y(t_n)$ at time t_n. The *region of absolute stability for a method* is the set of values of the time step h and of λ for which

$$\lim_{n \to \infty} y_n = \vec{0}$$

is verified. One can consider absolute stability as the capability of a method to bare brutal variations of the solution, even with large time steps. This is preponderant with real-time multiscale simulations, which induce the choice of optimal time steps.

In the multidimensional case given by equation (2), a necessary condition for the absolute stability of a method is that $h\lambda$ be in the stability region of this method for each eigenvalue λ of A and h the largest time step.

B. Examples of classical Euler methods

Before exposing what regards asynchronous schemes, we recall classical results about elementary methods. The simplest method for solving (2) is the Euler algorithm. It is given by

$$X_n = X_{n-1} + hA \cdot X_{n-1} \tag{3}$$

In the one dimensional case, one easily gets the absolute stability region : this is the open disk defined by $\{z = hA \in \mathbb{C} \ : \ |1+z| < 1\}$.
Consider the simple example $A = -\lambda$, $\lambda \in \mathbb{R}_+$. Equation (2) is simply $X'(t) = -\lambda X(t)$, and its solution is $X(t) = e^{-\lambda t} X(0)$. Applying explicit Euler scheme, one obtains $X_n = (1-\lambda h)^n X_0$, and the absolute stability condition is $|1-\lambda h| < 1$, that is, $h < 2/\lambda$. Figure 2 shows different approximations of the solution with $\lambda = 6$, i.e. $2/\lambda = 1/3$.

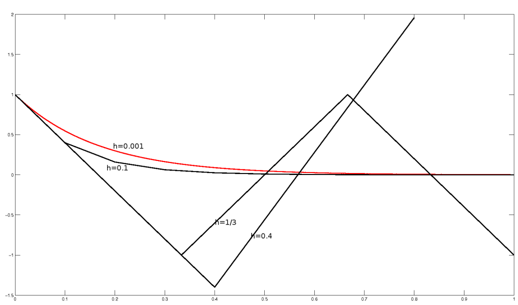

Fig. 2. Explicit Euler scheme applied to test-equation $y' = -6y$ with different time steps. The absolute stability threshold $h = 1/3$ is highlighted.

In the general multidimensional case, equation (3) gives

$$X_n = (I + hA)^n \cdot X_0 \tag{4}$$

where I is the identity matrix. Therefore (see section IV), the absolute stability condition is here

$$\rho(I + hA) < 1$$

with $\rho(M)$ the spectral radius of M.

A more efficient algorithm, regarding absolute stability, is given by the Implicit Euler method

$$X_n = X_{n-1} + hA \cdot X_n \tag{5}$$

Here, one easily gets the fact that the absolute stability region is the whole complex plane. Indeed, for the one dimensional test-equation, one gets with implicit Euler method $X_n = \frac{1}{(1+\lambda h)^n} X_0$, so that $\lim_{n\to\infty} X_n = 0 \ \forall h$, and absolute stability is guaranteed. Figure 3 shows different approximations, for the same example and the same values of the time step as in figure 2.

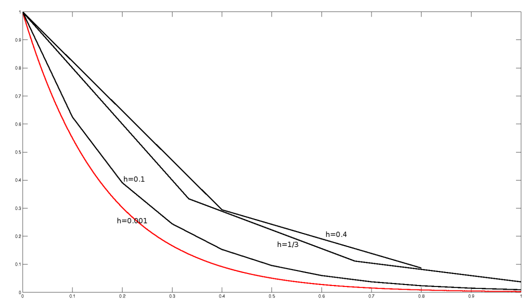

Fig. 3. An application of Implicit Euler scheme for the test-equation $y' = -6y$ with the same values of the time step as in figure 2.

III. Asynchronous and chaotic asynchronous schemes

Chaotic asynchronous schemes were presented in [3] and [4], where their general definition and convergence properties were detailed. For the sake of simplicity, and because we deal with absolute stability, we will simply remind the reader of the principle of asynchronous and chaotic asynchronous scheduling, when applied to test-equation (2). The example of explicit Euler scheme, though simple, will enable us to stress the difference between asynchronous and chaotic asynchronous schemes, so as problems posed by "poor" properties of the spectral radius.

Here is the principle: we consider equation (2) and assume that matrix A is written

$$A = \sum_{i=1}^{m} A_i \tag{6}$$

As regards applications in the domain of interactive real-time simulations, each A_i is the matricial representation of a distinct phenomenon. Each of these phenomena will be activated at specific moments. In the asynchronous case one defines a scheduling that will be repeated all along the simulation. In the chaotic asynchronous case, this order of phenomena activations changes randomly at each cycle.
The next sections describe in details these simulation methods.

A. Asynchronous Euler schemes

Consider a fixed permutation $\sigma \in S_m$, where S_m is the symmetric group of permutations of m elements. This permutation is used at each time step, and characterizes the scheduling of A_i's interventions in cycles. We recall that this "desynchronization" mainly makes it easy to add or remove phenomena in the course of a running simulation, without stopping it.

The principle is to execute the same algorithm (here explicit Euler) successively with each phenomenon involved, according to the order of interventions fixed by the permutation σ. On one time step, the execution of asynchronous explicit Euler algorithm gives :

$$\begin{aligned} X^{*1} &= X_{n-1} + hA_{\sigma(1)} \cdot X_{n-1} \\ X^{*2} &= X^{*1} + hA_{\sigma(2)} \cdot X^{*1} \\ &\vdots \\ X_n &= X^{*(m-1)} + hA_{\sigma(m)} \cdot X^{*(m-1)} \end{aligned}$$

Thus, one gets

$$X_n = \left(\prod_{i=1}^{m} (I + hA_{\sigma(i)}) \right)^n \cdot X_0 \tag{7}$$

We stress again the point that the same permutation σ is used here on each time step.
In a similar way, asynchronous scheme applied to implicit Euler algorithm leads to :

$$X_n = \left(\prod_{i=1}^{m} \left(I - hA_{\sigma(m-i+1)} \right)^{-1} \right)^n \cdot X_0 \tag{8}$$

B. Chaotic asynchronous explicit Euler scheme

The fundamental difference between asynchronous and chaotic asynchronous schemes is that a new permutation is chosen at each time step for the scheduling of phenomena. During time step n, the order of interventions of phenomena involved makes matrices intervene the following way : $A_{\sigma_n(1)}, A_{\sigma_n(2)}, \dots, A_{\sigma_n(m)}$, where σ_n is the permutation of m operators A_i which is involved at time n.

For this time step, chaotic asynchronous Euler algorithm gives :

$$\begin{aligned} X^{*1} &= X_{n-1} + hA_{\sigma_n(1)} \cdot X_{n-1} \\ X^{*2} &= X^{*1} + hA_{\sigma_n(2)} \cdot X^{*1} \\ &\vdots \\ X_n &= X^{*(m-1)} + hA_{\sigma_n(m)} \cdot X^{*(m-1)} \end{aligned}$$

Thus, one gets

$$X_n = \prod_{i=1}^{m} (I + hA_{\sigma_n(i)}) \cdot X_{n-1}$$

that is

$$X_n = \prod_{k=1}^{n} \prod_{i=1}^{m} (I + hA_{\sigma_k(i)}) \cdot X_0$$

Here again, this chaotic asynchronous scheme may be applied to implicit Euler algorithm and leads to

$$X_n = \prod_{k=1}^{n} \prod_{i=1}^{m} (I - hA_{\sigma_k(m-i+1)})^{-1} \cdot X_0$$

As an introduction to the kind of problems that arise when using these methods, the next part deals exclusively with the asynchronous case. The chaotic case will be even more difficult to handle, because it involves stochastic processes.

IV. Issues and results about asynchronous Euler schemes

In the following, we denote by $\rho(M)$ the spectral radius of a matrix M. We will make use of the following fundamental property:

Theorem IV.1. *[15] Let M a matrix in $\mathbb{C}^{n \times n}$.*

$$\lim_{n \to \infty} M^n = 0 \iff \rho(M) < 1$$

A. Stability regions

Considering equation (4), theorem IV.1 implies that the absolute stability condition for explicit Euler scheme is given by

$$\rho(I + hA) < 1 \tag{9}$$

The same way, considering equation (7), the absolute stability condition for asynchronous explicit Euler scheme is given by

$$\rho \left(\prod_{i=1}^{m} (I + hA_{\sigma(i)}) \right) < 1$$

with σ the fixed permutation chosen at the beginning of the execution. An obvious remark is that this condition is not as easy to check as (9), and may induce complex computations (our damped mass-spring example will exhibit this complexity). This is the reason why it is important to provide absolute stability conditions for these asynchronous schemes. This is what we present in the following.

Moreover, since any permutation may be initially chosen and then used during the whole simulation, we get the trivial following criteria for explicit and implicit asynchronous Euler schemes:

Proposition IV.2.
1) *The absolute stability domain for asynchronous explicit Euler scheme, when resolving $X' = A \cdot X = (\sum_{i=1}^{m} A_i) \cdot X$, is given by the set*

$$S_A = \{h \in \mathbb{R}_+ : \forall \sigma \in S_m, \rho(\prod_{i=1}^{m} (I + hA_{\sigma(i)})) < 1\}$$

2) *The absolute stability domain for asynchronous implicit Euler scheme, when resolving $X' = A \cdot X = (\sum_{i=1}^{m} A_i) \cdot X$, is given by the set*

$$S_A = \{h \in \mathbb{R}_+ : \forall \sigma \in S_m, \rho(\prod_{i=1}^{m} (I - hA_{\sigma(m-i+1)})^{-1}) < 1\}$$

In section VI, a detailed example will show that these criteria may induce complex conditions on time steps, when applied to concrete cases. But even the most simple case of a one dimensional equation leads to non trivial conditions, the following example may be instructive.

B. Examples of stability regions in one dimension

In this section we illustrate the non triviality of absolute stability conditions for asynchronous schemes, even in elementary cases. We want to show that the conditions of absolute stability for asynchronous schemes, in both cases of explicit and implicit Euler, are uneasy to handle in general. Even in the simple case of one differential equation, where all A_i are real numbers and commute, conditions on the spectral radius become $|\prod_{i=1}^{m}(1+hA_i)| < 1$ and $|\prod_{i=1}^{m}\frac{1}{1-hA_i}| < 1$, so that a general condition on h is not easy to extract. For instance, one can consider the special case where $m = 2$ and A_1, A_2 are real numbers, here denoted $-\lambda_1$ and $-\lambda_2$: we assume in the following $\lambda_1 + \lambda_2 > 0$, so that the problem

$$x'(t) = -(\lambda_1 + \lambda_2)x(t), \quad \lambda_1 + \lambda_2 > 0 \tag{10}$$

remains stiff.

In the case of the explicit Euler scheme, the absolute stability condition for (10) is $|(1 - \lambda_1 h)(1 - \lambda_2 h)| < 1$. A direct study leads to the following alternative:

Proposition IV.3.
- *If $\lambda_1\lambda_2 > 0$, the absolute stability condition for (10) is*

$$h \in \left]0; \frac{\lambda_1 + \lambda_2 - \sqrt{(\lambda_1+\lambda_2)^2 - 8\lambda_1\lambda_2}}{2\lambda_1\lambda_2}\right[\cup \left]\frac{\lambda_1 + \lambda_2 + \sqrt{(\lambda_1+\lambda_2)^2 - 8\lambda_1\lambda_2}}{2\lambda_1\lambda_2}; \frac{\lambda_1+\lambda_2}{\lambda_1\lambda_2}\right[$$

- *If $\lambda_1\lambda_2 < 0$, the absolute stability condition for (10) is*

$$h \in \left]0; \frac{\lambda_1 + \lambda_2 - \sqrt{(\lambda_1+\lambda_2)^2 - 8\lambda_1\lambda_2}}{2\lambda_1\lambda_2}\right[$$

On the other hand, the absolute stability condition for the implicit scheme (5) is $\left|\frac{1}{(1+h\lambda_1)(1+h\lambda_2)}\right| < 1$ what leads to another alternative:

Proposition IV.4.
- *If $\lambda_1\lambda_2 > 0$, the absolute stability condition for (10) is trivial, so that the method is absolutely stable.*
- *If $\lambda_1\lambda_2 < 0$, the absolute stability condition for (10) is*

$$h \in \left]0; \frac{\lambda_1+\lambda_2}{-\lambda_1\lambda_2}\right[\cup \left]\frac{\lambda_1 + \lambda_2 + \sqrt{(\lambda_1+\lambda_2)^2 - 8\lambda_1\lambda_2}}{-2\lambda_1\lambda_2}; +\infty\right[$$

For example, let us consider the case $\lambda_1 = -3,\ \lambda_2 = 11$, so that $\lambda = \lambda_1 + \lambda_2 = 8$. Therefore our problem is the stiff one :

$$x' = -8x = -(-3 + 11)x$$

Absolute stability conditions are in this case :
- classical synchronous Euler : $h < \frac{2}{8} = 0.25$
- explicit asynchronous Euler: $h < 0.1531$
- implicit asynchronous Euler :

$$h \in]0; 0.2424[\cup]0.3956; +\infty[\tag{11}$$

One can check these results with different simulations.

This simple example suggests that the exact absolute stability region of a general asynchronous scheme may be really complex. Nevertheless, we can prove an easier-to-apply (but less precise) criterion for the explicit case.

C. Criterion of absolute stability (explicit scheme)

The proof of the following proposition is based on majoration of $\| \prod_{i=1}^{m}(I_n + hA_\sigma(i))\|$. Details can of course be asked to authors.

Proposition IV.5. *Consider the decomposition $A = \sum_{i=1}^{m} A_i$ where $A \in \mathcal{L}(\mathbb{C}^n)$. Let P be the passage matrix into a base where A is triangular, and the norm defined by*

$$\|v\|_A = \|P^{-1}v\|_1$$

Let $M = \max_i \|A_i\|_A$, and $P_m(X)$ the polynomial defined by $P_m(X) = (X+1)^m - 1 - mX$. Then, the absolute stability region for Euler chaotic asynchronous scheme, with the desynchronization considered, contains the set

$$S_A = \{h > 0 : \ 0 < \rho(I_n + hA) < 1 - P_m(hM)\}.$$

We will see an illustration of this criterion in section VI. But for now, in the next section, we study the chaotic asynchronous case.

V. Chaotic asynchronous Euler schemes

This section presents our main results about absolute stability of chaotic asynchronous schemes. We recall that the fundamental difference between asynchronous and chaotic asynchronous schemes is the fact that, in the latter case, a new permutation is chosen at each time step for the scheduling of phenomena. This leads to radically different properties of stability, as detailed below.

Let us first recall that the execution of chaotic asynchronous Euler schemes, when solving equation (2) with the decomposition (6), leads to the following formulas, where σ_k is the permutation used at step k, $k \leq n$:

- For chaotic asynchronous explicit Euler scheme,

$$X_n = \prod_{k=1}^{n}\prod_{i=1}^{m}(I + hA_{\sigma_k(i)}) \cdot X_0$$

- For chaotic asynchronous implicit Euler scheme,

$$X_n = \prod_{k=1}^{n}\prod_{i=1}^{m}(I - hA_{\sigma_k(m-i+1)})^{-1} \cdot X_0$$

In this section, we prove a general criterion which ensures that the upper Lyapunov exponent associated with a distribution on $GL(d, \mathbb{R})$ is negative. Then we apply this criterion to the absolute stability of chaotic asynchronous methods.

A. Negative upper Lyapunov exponent

Let us start with some common notations and definitions (see [16] for a detailed theory about the products of random matrices).

Definition V.1. *If $(B_i)_{i\geq 1}$ is a sequence of i.i.d random matrices, we write β_n the product $B_n.\cdots.B_1$. If $\ln^+\|B_1\|$ is integrable, then the following limit exists and is called upper Lyapunov exponent of the sequence (or equivalentely of the distribution associated to the sequence):*

$$\lim_n \frac{1}{n}\mathbb{E}\left[\ln\|\beta_n\|\right] = \gamma$$

Definition V.2. *If μ is a probability measure on $GL(d,\mathbb{R})$, G_μ is the smallest closed subgroup of $GL(d,\mathbb{R})$ that contains the support of μ.*

Definition V.3. *A subset S of $GL(d,\mathbb{R})$ is said to be irreducible if there is no proper subspace $V\subset\mathbb{R}^d$ such that $M(V)=V$ for all $M\in S$.*

We will use the following lemma:

Lemma V.4. *Let $\{B_n, n\geq 1\}$ be a sequence of independent random matrices of $GL(d,\mathbb{R})$ with common distribution μ, and $\beta_n = B_n.\cdots.B_1$. We suppose that:*

1) *G_μ is irreducible.*
2) *$\ln^+\|B_1\| + \ln^+\left\|B_1^{-1}\right\|$ is integrable*

Then

$$\lim_n \frac{1}{n}\sup_{\|x\|=1}\mathbb{E}\left[\ln\|\beta_n\cdot x\|\right] = \gamma$$

Proof: First of all, let us check that the sequence

$$a_n = \sup_{\|x\|=1}\mathbb{E}\left[\ln\|\beta_n\cdot x\|\right]$$

is subadditive. For any integer n and m one has

$$\begin{aligned}\mathbb{E}\left[\ln\|\beta_{n+m}.x\|\right] &= \mathbb{E}\left[\ln\|B_{n+m}\cdots B_{n+1}B_n\cdots B_1\cdot x\|\right]\\ &= \mathbb{E}\left[\ln\left\|B_{n+m}\cdots B_{n+1}\frac{B_n\cdots B.x}{\|B_n\cdots B_1\cdot x\|}\right\|\right]\\ &\quad + \mathbb{E}\left[\ln\|B_n\dots B_1\cdot x\|\right]\end{aligned}$$

As $\dfrac{B_n\cdots B_1\cdot x}{\|B_n\cdots B_1\cdot x\|}$ is unitary, considering the upper bounds on $\|x\|=1$ leads to

$$a_{n+m}\leq a_n + a_m$$

Thereby, the sequence $\frac{a_n}{n}$ converges: we denote γ' its limit.

Since G_μ is irreducible and $\mathbb{E}\left[\ln^+\|B_1\|\right]$ is finite, we know that, for any $x\neq 0$ (see [16] p.72):

$$\lim_n \frac{1}{n}\ln\|\beta_n\cdot x\| = \gamma \quad \text{almost surely.}$$

Now, an easy computation shows that

$$\frac{1}{n}\left|\ln\|\beta_n\cdot x\|\right| \leq \frac{1}{n}\sum_{i=1}^n\left(\ln^+\|B_i\| + \ln^+\left\|B_i^{-1}\right\|\right)$$

From the law of large numbers, the right hand side converges in L^1, thereby, it is uniformly integrable. Thus, the left hand side is also uniformly integrable, and as it converges almost surely to γ, it converges as well in L^1:

$$\lim_n \frac{1}{n}\mathbb{E}\left[\ln\|\beta_n.x\|\right] = \gamma \tag{12}$$

And since $\dfrac{1}{n}\mathbb{E}\left[\ln\|\beta_n\cdot x\|\right]\leq\dfrac{a_n}{n}$, one has

$$\gamma\leq\gamma'$$

On the other hand, one has

$$\sup_{\|x\|=1}\mathbb{E}\left[\ln\|\beta_n\cdot x\|\right]\leq\mathbb{E}\left[\sup_{\|x\|=1}\ln\|\beta_n\cdot x\|\right] = \mathbb{E}\left[\ln\|\beta_n\|\right]$$

The right hand side, by definition, converges to γ, so that $\gamma'\leq\gamma$, which ends the proof. ■

From this lemma, one can deduce the following result based on the negativity of the upper Lyapunov exponent:

Proposition V.5. *Let $\{B_n, n\geq 1\}$ be a sequence of independent random matrices of $GL(d,\mathbb{R})$ with common distribution μ that satisfies*

1) *G_μ is irreducible.*
2) *$\ln^+\|B_1\| + \ln^+\left\|B_1^{-1}\right\|$ is integrable*

If there exists an integer m such that

$$\sup_{\|x\|=1}\mathbb{E}\left[\ln\|\beta_m\cdot x\|\right] < 0$$

Then, for any x

$$\lim_n \beta_n\cdot x = 0 \quad \textit{almost surely.}$$

Proof: From lemma V.4, we know that

$$\lim_n \frac{1}{n}\sup_{\|x\|=1}\mathbb{E}\left[\ln\|\beta_n\cdot x\|\right] = \gamma$$

But, as $a_n = \sup_{\|x\|=1}\mathbb{E}\left[\ln\|\beta_n\cdot x\|\right]$ is a subadditive sequence, we know that

$$\lim_n \frac{a_n}{n} = \inf_m \frac{a_m}{m}$$

Our hypothesis ensures that $\inf_m \dfrac{a_m}{m} < 0$, so that $\gamma < 0$. But, as in lemma V.4, we know that for any $x\neq 0$

$$\lim_n \frac{1}{n}\ln\|\beta_n\cdot x\| = \gamma \quad \text{almost surely.}$$

This suffices to deduce the result. ■

B. Absolute stability of chaotic asynchronous schemes

In this section we will simply apply proposition V.5 to chaotic asynchronous schemes. In this particular context, assumptions of this proposition are generally satisfied, so that the following criterion is relevant.

Definition V.6. *In the following proposition, a matrix is said to be associated with a chaotic asynchronous method if it is a random product of matrices intervening at each time step: for instance, matrices associated with chaotic asynchronous* *explicit euler scheme for the resolution of $X' = (\sum_{i=1}^m A_i)\cdot X$ will be the following products:*

$$B_k = \prod_{i=1}^{m} (I + hA_{\sigma_k(i)}),\ \sigma_k \in S_m$$

Of course, our problem regards the limit of products of such associated matrices.

Proposition V.7. *Let $B = \{B_1, \ldots, B_N\} \subset GL(d, \mathbb{R})$ be the matrices associated with a chaotic asynchronous method applied to a linear equation. We suppose that B is irreducible, then, if there exists an integer m such that*

$$\sup_{\|x\|=1} \prod_{1\leq, i_1, \cdots, i_m \leq N} \|B_{i_1} \cdots B_{i_m} \cdot x\| < 1$$

Then the method is almost surely absolutely stable.

Proof: First, it is easy to check that if B is irreducible, then G_μ is also irreducible (where μ is the uniform distribution on B).

Since the matrices are equidistributed, one has:

$$\begin{aligned}
&\sup_{\|x\|=1} \mathbb{E}\left[\ln \|\beta_m \cdot x\|\right] \\
&\quad = \sup_{\|x\|=1} \tfrac{1}{N^m} \textstyle\sum_{1\leq, i_1, \ldots, i_m \leq N} \ln \|B_{i_1} \cdots B_{i_m} \cdot x\| \\
&\quad = \sup_{\|x\|=1} \tfrac{1}{N^m} \ln \left(\textstyle\prod_{1\leq, i_1, \ldots, i_m \leq N} \|B_{i_1} \cdots B_{i_m} \cdot x\| \right) \\
&\quad = \tfrac{1}{N^m} \ln \left(\sup_{\|x\|=1} \textstyle\prod_{1\leq, i_1, \ldots, i_m \leq N} \|B_{i_1} \cdots B_{i_m} \cdot x\| \right)
\end{aligned}$$

Our hypothesis insures that

$$\sup_{\|x\|=1} \mathbb{E}\left[\ln \|\beta_m \cdot x\|\right] < 0$$

Since we have only a finite number of matrices, the condition of integrability of $\ln^+ \|B_1\| + \ln^+ \left\|B_1^{-1}\right\|$ is satisfied. Thus we may apply proposition V.5 and conclude. ■

With quite simple calculus this criterion can indicate, depending on the value of h, that a chaotic asynchronous method is stable. Nevertheless in some cases, the criteria is not applicable because the sequence of functions $\prod_{1\leq, i_1, \cdots, i_m \leq N} \|B_{i_1} \cdots B_{i_m} \cdot x\|$ converges only almost everywhere. We could have improved the criterion to handle this fact, and produce a result like the following one :

Proposition V.8. *Let $S^d = \{x \in \mathbb{R}^d, \|x\| = 1\}$. If there exists an integer m and a subset $N \subset S^d$ of null measure such that*

$$\sup_{x\in S-N} \prod_{1\leq, i_1, \cdots, i_m \leq N} \|B_{i_1} \cdots B_{i_m} \cdot x\| < 1$$

Then the method is almost surely absolutely stable.

But such a proposition would be useless in practice. Even in the case of a negative Liapunov exponent, the quantity $\prod_{1\leq, i_1, \cdots, i_m \leq N} \|B_{i_1} \cdots B_{i_m} \cdot x\|$ may grow to infinity on a set of null measure (this is precisely the case of system S_1 in section VI-E). In these cases, the estimation of Lyapunov exponent may become the only way to compute stability conditions.

The next section is devoted to examples and illustrations of all the previous results and observations.

VI. Applications and illustrations

The damped mass-spring system is a particular case of desynchronisation of one single differential equation, this is why we first describe this general case.

A. Desynchronization of one single differential equation

The case of one single linear differential equation with order m is given by:

$$x^{(m)}(t) - \sum_{i=0}^{m-1} a_i x^{(i)}(t) = 0 \tag{13}$$

This equation can be written as a linear differential system: with the notations $z_i = x^{(i)},\ 0 \leq i \leq m-1$, one gets the system

$$\begin{pmatrix} z_0 \\ z_1 \\ \vdots \\ z_{m-1} \end{pmatrix}' = \begin{pmatrix} 0 & 1 & 0 & \ldots \\ 0 & 0 & \ddots & 0 \\ 0 & \ldots & 0 & 1 \\ a_0 & a_1 & \ldots & a_{m-1} \end{pmatrix} \cdot \begin{pmatrix} z_0 \\ z_1 \\ \vdots \\ z_{m-1} \end{pmatrix}$$

what can be denoted, with obvious notations,

$$Z' = A \cdot Z.$$

Consider the elementary matrices $E_{ij},\ 1 \leq i, j \leq m$. Assume that coefficients a_i each characterize a distinct phenomenon: we can associate to a_i the matrix

$$P_i = a_i E_{m,i+1}$$

and introduce an "integration phenomenon" given by the matrix

$$\text{Int} = \sum_{i=1}^{m-1} E_{i,i+1}$$

With these notations, one easyly gets

$$A = \text{Int} + \sum_{i=0}^{m-1} P_i$$

Therefore, one can apply an asynchronous scheme (chaotic or not), where the P_is and the integration phenomenon are desynchronized. Our main example of a damped mass-spring system will illustrate this process.

B. Damped mass-spring system

In sections IV and V, we have exposed absolute stability conditions for asynchronous and chaotic asynchronous schemes. In the following, we propose an illustration of these results in the case of a second order linear differential equation, with drastic absolute stability conditions when physical phenomena involved are desynchronized. We volunteerly consider a typical case of antagonist phenomena leading to a more significative unstability when they are desynchronized. Indeed, we consider the case of a damped spring-mass system, that can be represented by the following equation

$$x'' = -g - \frac{k}{m}x - \frac{\gamma}{m}x' \tag{14}$$

where:

- g is the gravity field
- m is the mass of the object
- k is the elasticity constant of the spring
- γ is the damp coefficient

All along this section, we will carry simple computations in order to illustrate our problems. We will consider two cases of such systems, defined by the following parameters :

$$(\gamma, k, m) = (1, 4, 1) \text{ referred as } (S_1) \tag{15}$$

and

$$(\gamma, k, m) = (8, 1, 1) \text{ referred as } (S_2) \tag{16}$$

But for now, we will try to explore our system in the general case. According to our theoretical study, we will first deal with the asynchronous case, before dealing with the chaotic asynchronous one. This example will clearly expose how chaotic schemes, though they are a bit more precise than non chaotic ones, may suffer from great instability.

Using the notations

$$x_1 = x, \quad x_2 = x'$$

equation (14) can be written as the system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ -k/m & -\gamma/m \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -g \end{bmatrix}$$

The simple change of variables

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ -k/m & -\gamma/m \end{bmatrix}^{-1} \cdot \begin{bmatrix} 0 \\ -g \end{bmatrix}$$

leads to the equivalent system

$$X' = \begin{bmatrix} 0 & 1 \\ -k/m & -\gamma/m \end{bmatrix} \cdot X$$

In the following, we use the notations:

$$A = \begin{bmatrix} 0 & 1 \\ -k/m & -\gamma/m \end{bmatrix}, \ A_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 0 \\ -k/m & 0 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 0 & 0 \\ 0 & -\gamma/m \end{bmatrix}$$

so that $A = A_1 + A_2 + A_3$.

Our study of absolute stability implies that the eigenvalues of A both be in the negative half-plane. Since these eigenvalues are

$$\lambda_\pm = \frac{-\gamma \pm \sqrt{\gamma^2 - 4mk}}{2m}$$

a direct computation shows that $Re(\lambda_\pm) < 0 \iff (k, \gamma) \in (\mathbb{R}_+^*)^2$

Before exposing results for asynchronous schemes, we first recall classical results as regards equation (14) in the case of classical Euler schemes.

C. Classical Euler schemes

First, in the explicit Euler case, the absolute stability condition is here $\rho(I + hA) \in [0; 1[$, and is equivalent to $|1 + h\lambda_\pm| < 1$. Therefore, we get the conditions :

- If $\gamma^2 - 4mk \geq 0$,
$$h < \frac{4m}{\gamma + \sqrt{\gamma^2 - 4mk}}.$$
- If $\gamma^2 - 4mk \leq 0$,
$$h < \frac{\gamma}{k}.$$

  In the case of system (S_1) the condition is $h < 0.25$ and for (S_2) one gets $h < 8 - 2\sqrt{15} \sim 0.254$

In the implicit case, the absolute stability condition is $\rho((I - hA)^{-1}) < 1$. Nevertheless, this condition is trivial, since we have seen that implicit Euler method is absolutely stable, with no condition on the time step. This can obviously be verified by considering eigenvalues of $(I - hA)^{-1}$.

D. Absolute stability conditions for asynchronous Euler schemes

Now we turn to asynchronous methods and we will show how conditions given in proposition IV.2, though simple, can lead to difficult computations, even on our elementary example.

Notice that proofs of propositions VI.1 and VI.2 are both based on quite technical computations, especially for the study of multiple-parameters 4th degree polynomials. Details of these proofs can be asked to authors.

1) Asynchronous explicit Euler: We prove the following result :

Proposition VI.1. *Conditions of absolute stability for asynchronous explicit Euler scheme for the damped mass-spring system are the following :*

- *If* $2\gamma^2 - k < 0$,
$$h < -\frac{\gamma}{m} + \sqrt{\frac{\gamma^2}{m^2} + 4\frac{m}{k}} \tag{17}$$
- *If* $2\gamma^2 - k \geq 0$,
$$h < \frac{m}{\gamma} \tag{18}$$

Once again we will illustrate these results with our two systems (S_1) and (S_2). In the first case, the condition is $h < \sqrt{2} - 1 \sim 0.4142$, and in the second case $h < 0.125$. Moreover, we computed the criterion given in section IV.5. The following table summarizes all these results :

System	Classic	Asynchronous	Criterion
S_1	0.25	0.414	0.084
S_2	0.254	0.125	0.061

TABLE I

COMPARISON OF STABILITY CONDITIONS ON h FOR EXPLICIT METHODS. THE FIRST COLUMN SHOWS THE STABILITY CONDITIONS FOR THE CLASSIC EXPLICIT METHOD, THE SECOND COLUMN SHOWS THE EXACT CONDITIONS IN THE ASYNCHRONOUS CASE AND THE THIRD ONE SHOWS THE CONDITION BASED ON THE PROPOSITION IV.5.

One can notice that in the case of system (S_1), asynchronous explicit Euler scheme gives better results than the classical scheme. Moreover, the criterion given in proposition IV.5 is quite easy to use, but gives strong majorations.

2) Asynchronous implicit Euler: We prove the following result :

Proposition VI.2. *A sufficient condition for absolute stability of asynchronous implicit euler scheme, in the case of a damped spring-mass system, is*

$$h < \frac{\tau}{\sqrt{k}} \tag{19}$$

with τ the biggest real positive root of the polynomial

$$\Delta(X) = \alpha X^3 + X^2 - 2\alpha X - 4, \quad \alpha = \frac{\gamma}{\sqrt{mk}}$$

Remark : A good approximation of $h < \tau$ is given by

$$h < \frac{m}{2\gamma}\left(-1 + 2\frac{\gamma}{\sqrt{mk}} + \sqrt{1 + 4\frac{\gamma}{\sqrt{mk}}}\right)$$

There is no need to explore our systems (S_1) and (S_2) according to implicit asynchronous method. Indeed, implicit Euler scheme is absolutely stable, but from the previous result, we know that asynchronous implicit Euler scheme is not stable (for any value of h). This illustrates clearly the loss of performance of this method.

E. Absolute stability of chaotic asynchronous Euler schemes

We finally illustrate our theoretical results for the damped spring-mass system with our systems (S_1) and (S_2). First, we illustrate the complex behavior of the Lyapunov exponent, and the fact that it need not be better from explicit scheme to implicit ones. In each of the cases exposed on figures 4 and 5, we compute numerically (using approximations of invariant measures) the Lyapunov exponents in function of the time step.

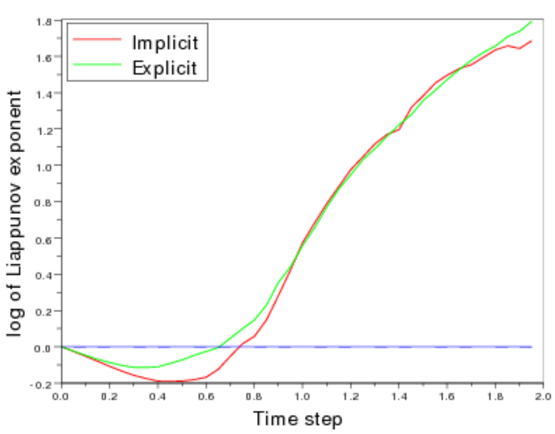

Fig. 4. comparison of chaotic asynchronous explicit and implicit Euler, system S_1

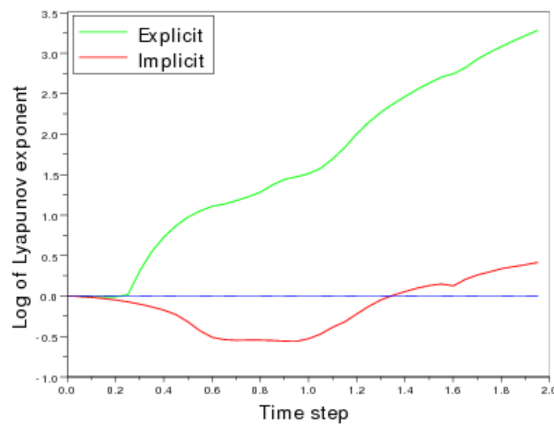

Fig. 5. comparison of chaotic asynchronous explicit and implicit Euler, system S_2

These two figures show that the behavior of the upper Lyapunov exponent does not make implicit chaotic schemes more stable than the explicit ones, unlike in the classical Euler schemes. Any case may occur.

To end with, we computed the values of the different criteria from proposition V.7, for different values of m. The following table summarizes the calculus.

System, scheme	Lyapunov exp.	Crit. $m = 2$	Crit. $m = 3$
S_1 explicit	0.728	—	—
S_1 implicit	0.652	—	—
S_2 explicit	0.227	0.208	0.217
S_2 implicit	1.341	1.257	1.285

TABLE II
STABILITY CONDITIONS. THE FIRST COLUMN CORRESPOND TO CONDITIONS ON h COMPUTED FROM THE ESTIMATION OF LYAPUNOV EXPONENT, THE SECOND AND THIRD ONES GIVE CONDITIONS FROM THE APPLICATION OF THE PROPOSITION V.7 WITH $m = 2$ AND $m = 3$.

This table shows that system S_1 is an example of a situation where proposition V.7 does not apply, as the sequence $\sup_{\|x\|=1} \prod_{1 \leq, i_1, \cdots, i_m \leq N} \|B_{i_1} \cdots B_{i_m} \cdot x\|$ does not converge quickly enough. On the other hand, with system S_2, one can easily compute conditions on h without estimating the Lyapunov exponent.

These simple examples exhibit the fact that a systematic application of chaotic asynchronous methods leads to quite unpredictable systems, as regards their absolute stability.

VII. Conclusion and Proposition

Chaotic asynchronous schemes for resolving ordinary differential systems have shown their interest in the context of real time interactive simulation of multi interaction systems, especially when dealing with biochemical kinetics. Their main advantage is the capability that is given to the user to add or remove interactions, e.g. chemical reactions or forces, in the course of a simulation. Nevertheless, eventhough proofs of convergence for such schemes have been established, the present work highlights the fact that absolute stability conditions may be difficult to satisfy, when antagonist phenomena are desynchronized: antagonist forces can lead to force the choice of tiny time steps, making impossible the aim of real-time simulation. An illustration is given by the case of a mass-spring system. Therefore, a compromise has to be found between a total desynchronization of phenomena, which leads to instability, and synchronization, which prevents from in virtuo experimentation.

We propose, in this perspective, to adapt splitting methods [17] in order to keep the capacity of interacting by adding or removing phenomena. Indeed, splitting methods seem to be relevant when the phenomena involved in a simulated system have to be considered as autonomous: as in the case of chaotic asynchronous schemes, the resolution of a system $y' = (A+B)y$ is replaced by successive resolutions of systems $y' = Ay$ and $y' = By$. The use of different time steps for each of the subsystems permits to simulate multiscale systems. This is also possible with chaotic asynchronous schemes, but splitting methods have the advantage of absolute stability, by the use of particular scheduling of integrations of each subsystem, each of which being solved by an absolutely stable method. Nevertheless, the choice of splitting methods makes it impossible to add or remove phenomena in the course of a simulation, without stopping the simulation and rewriting algorithms with the new set of phenomena involved.

We have recently developed algorithms that can be seen as an hybridation between chaotic asynchronous schemes and splitting methods: a future work will expose these methods and achieve their theoretical study.

AKNOWLEDGEMENT

The work exposed here receives a financial support from the French National Research Agency (ANR) within the program Complex Systems and Mathematical Modelling (SYSCOMM): ANR-08-SYSC-002

REFERENCES

[1] G. Desmeulles, S. Bonneaud, P. Redou, V. Rodin, and J. Tisseau, "In Virtuo Experiments Based on the Multi-Interaction System Framework: the RéISCOP Meta-Model," *CMES: Computer modeling in engineering and sciences*, vol. 47, no. 3, pp. 299–330, 2009.

[2] J. Tisseau, "Virtual Reality -in virtuo autonomy-," *Accreditation to Direct Research*, 2001.

[3] P. Redou, S. Kerdélo, C. Le Gal, G. Querrec, V. Rodin, J.-F. Abgrall, and J. Tisseau, "Reaction-Agents: First Mathematical Validation of a Multi-Agent System for Dynamical Biochemical Kinetics," *Progress in Artificial Intelligence (EPIA 2005), Lecture Notes in Artificial Intelligence, Springer*, 2005.

[4] P. Redou, G. Desmeulles, J. Abgrall, V. Rodin, and J. Tisseau, "Formal validation of asynchronous interaction-agents algorithms for reaction-diffusion problems." PADS'07, 21st International Workshop on Principles of Advanced and Distributed Simulation, june 2007.

[5] R. L. Devaney, *An Introduction to Chaotic Dynamical Systems, 2nd ed.* Westview Press, 2003.

[6] D. Hanahan and J. Weinberg, "The Hallmarks of Cancer," *Cell*, vol. 100, pp. 57–70, 2000.

[7] J. Bos, "Ras Oncogene in Human Cancer : a Review," *Cancer Research*, vol. 50, pp. 1352–1361, 1989.

[8] V. Rodin, G. Querrec, P. Ballet, R. Bataille, G. Desmeulles, J. Abgrall, and J. Tisseau, "Multi-Agents System to Model Cell Signalling by Using Fuzzy Cognitive Maps. Application to Computer Simulation of Multiple Myeloma." Proceedings of the 9th IEEE International Conference on Bioinformatics and Bioengineering (BIBE'09), june 2009.

[9] G. Querrec, V. Rodin, J. Abgrall, S. Kerdelo, and J. Tisseau, "Uses of Multiagents Systems for Simulation of MAPK Pathway." Third IEEE Symposium on BioInformatics and BioEngineering, 2003, pp. 421–425.

[10] G. Lu, G. Broze, and S. Krishnaswamy, "Formation of Factors IXa and Xa by the Extrinsic pathway. Differential Regulation by Tissue Factor Pathway Inhibitor and Antithrombin III," *The journal of biological chemistry*, vol. 279, no. 17, pp. 17 241–17 249, 2004.

[11] C. Le Gal, M. Parenthoen, P.-A. Béal, and J. Tisseau, "Comparison of Sea State Statistics Between a Phenomenological Model and Field Data." Aberden, Scotland: OCEANS 2007, June 2007.

[12] U. Ascher and L. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Siam, 1998.

[13] E. Hairer, S. Norsett, and P. Wanner, *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*. Springer Series in Comput. Mathematics, 1996.

[14] ——, *Solving Ordinary Differential Equations I. Nonstiff Problems*. Springer Series in Comput. Mathematics, 1983.

[15] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical Mathematics*. Springer, 2000.

[16] P. Bougerol and J. Lacroix, "Products of Random Matrices with Applications to Schrodinger Operators," *Progr. Probab. Statist.*, vol. 8, 1985.

[17] R. Mclachlan and G. Quispel, "Splitting Methods," *Acta Numerica*, vol. 11, pp. 341–434, 2002.